

## Desenvolvimento de uma aplicação para a emissão automatizada de certificados \*

Ana Clara B. Bordin<sup>1</sup>, Enzo Z. Becker<sup>1</sup>, Guilherme Moisés R. Garcia<sup>1</sup>,  
Leonardo Silva V. M. Barbosa<sup>1</sup>, Patrícia Pitthan Barcelos<sup>1</sup>

<sup>1</sup>Centro de Tecnologia - Universidade Federal de Santa Maria (UFSM)  
Santa Maria, RS – Brazil

{acbordin, ezbecker, gmgarcia, lsbarbosa, pitthan}@inf.ufsm.br

**Abstract.** *This article explores the implementation methodology and the operation of a certificate generator application, developed in the context of the Tutorial Education Program in Information Systems of the Federal University of Santa Maria. The tool was developed using Javascript programming language libraries that encompass file conversion and document encryption. The development of the tool resulted in a significant improvement in the production of certificates, which are now generated automatically.*

**Resumo.** *Este artigo explora a metodologia de implementação e o funcionamento de uma aplicação para emissão automatizada de certificados, desenvolvida no contexto do Programa de Educação Tutorial em Sistemas de Informação da Universidade Federal de Santa Maria. A ferramenta foi desenvolvida utilizando de bibliotecas da linguagem de programação Javascript que englobam a conversão de arquivos e a criptografia de documentos. O desenvolvimento da ferramenta resultou em uma melhoria significativa na produção de certificados, os quais passaram a ser gerados de forma automatizada.*

### 1. Introdução

O Programa de Educação Tutorial (PET) tem como objetivo aprimorar os cursos de graduação, atuando nos eixos de ensino, pesquisa e extensão. Os grupos PET realizam atividades em cada eixo, conforme definido em um planejamento anual. Por meio de suas atividades, o grupo PET Sistemas de Informação (PET-SI) da Universidade Federal de Santa Maria (UFSM) estimula a integração entre estudantes e docentes do curso de Sistemas de Informação, estudantes dos demais cursos da Instituição, estudantes de escolas de nível fundamental e nível médio e a comunidade.

Dentre as tarefas envolvidas na execução das atividades do grupo PET-SI, destacam-se: a realização de reuniões para estabelecer as metas e organizar o cronograma de cada atividade; a distribuição das tarefas entre os envolvidos na atividade; a organização da atividade, incluindo a alocação de espaço físico e de recursos; o contato com profissionais e convidados; a divulgação da atividade nos principais meios de comunicação, como página web, redes sociais e listas de e-mail; a execução efetiva da atividade; a elaboração e o envio de certificados aos participantes; a avaliação da atividade; e a elaboração de um relatório.

---

\*Trabalho apoiado pelo Programa de Educação Tutorial (PET/MEC).

Este artigo contempla uma forma de otimização de uma das tarefas pertinentes ao grupo PET-SI: a elaboração e o envio de certificados aos participantes das atividades. Os certificados de participação desempenham um papel crucial na formação dos alunos, uma vez que atuam como forma de reconhecimento e validação dos conhecimentos adquiridos durante a trajetória dos acadêmicos. No entanto, a confecção desses certificados, especificamente para as atividades realizadas pelo PET-SI da UFSM, costumava ser um processo totalmente manual, implicando em um consumo desnecessário de tempo e recursos.

Nesse contexto, a automatização do processo de emissão de certificados surge como uma alternativa para minimizar os esforços do grupo. Por meio do uso de tecnologias e ferramentas, foi possível criar uma aplicação que lê dados de um arquivo no formato CSV (*Comma Separated Values*) e os transforma em certificados individuais. A implementação da aplicação envolveu as etapas de projeto e desenvolvimento, visando uma gestão mais eficiente das atividades oferecidas pelo grupo PET-SI.

Este trabalho está organizado da seguinte forma: a Seção 2 apresenta um trabalho desenvolvido pelo grupo como primeira tentativa de resolver o problema de emissão de certificados. A Seção 3 exibe a motivação para o desenvolvimento de uma nova aplicação. Já a Seção 4 descreve a metodologia de desenvolvimento da aplicação. Na Seção 5 são expostos os resultados obtidos, enquanto a Seção 6 aponta as considerações finais acerca do trabalho realizado.

## **2. Revisão Bibliográfica**

Este segmento descreve um trabalho correlato, no qual o grupo tentou resolver o problema de emissão automatizada de certificados. O estudo foi conduzido por um membro do grupo, que desenvolveu uma aplicação que emprega a biblioteca Microsoft Office Interop Word para manipular documentos do Microsoft Word [Cassanego 2020].

A aplicação em questão tinha a capacidade de abrir um documento pré-existente, o modelo de certificado, e editá-lo por meio de linhas de código. Essa tarefa era realizada utilizando o método de busca e substituição, a fim de preencher os campos de texto. Embora essa estratégia tenha automatizado uma parte do processo, a aplicação ainda dependia de uma intervenção manual. Como consequência, o processo estava sujeito a erros de digitação, além de consumir um tempo significativo. Dessa forma, o grupo iniciou o projeto de uma aplicação para a emissão automatizada de certificados.

## **3. Justificativa e Motivação**

Um dos entraves na implementação de uma aplicação para emissão de certificados é o fácil acesso, por parte dos usuários, aos modelos e elementos usados na confecção dos mesmos, como é o caso da assinatura do emissor. Assim, buscou-se alternativas voltadas à criptografia de documentos. Neste contexto, foram realizados estudos sobre bibliotecas e recursos da linguagem de programação Javascript [cur 2022]. Inicialmente, foi preciso encontrar uma maneira de incorporar uma imagem de forma encriptada, garantindo assim que somente o proprietário da assinatura tivesse acesso a mesma. Para atender a este propósito, foi escolhida a biblioteca *crypto* da linguagem Javascript, a qual oferece operações de criptografia [cry 2019]. Essa implementação foi realizada de modo a assegurar que a imagem permanecesse criptografada na plataforma GitHub, exigindo uma chave privada para a sua descriptografia.

## 4. Metodologia

O desenvolvimento da aplicação foi dividido em quatro etapas, conforme descrito nas seções seguintes.

### 4.1. Coleta de dados

A coleta dos dados necessários para a aplicação ocorre por meio da ferramenta Google Formulários, um recurso usado para a criação e o gerenciamento de pesquisas. Ao término de cada atividade, os participantes são instruídos a escanear um código QR que os direciona para um formulário. Neste, são solicitados a fornecer seu nome completo, e-mail e número de matrícula. Essas informações servirão de base para a criação do conteúdo dos certificados.

### 4.2. Preparação dos dados

Após a etapa de coleta de dados, o responsável pela geração dos certificados acessa o formulário da atividade para a qual deseja emitir os certificados e faz o download do arquivo no formato CSV (*Comma Separated Values*) que contém as respostas do formulário. Um arquivo CSV é um arquivo de texto simples que armazena informações de planilhas e tabelas. O conteúdo de um arquivo CSV pode consistir em uma tabela de texto, números ou datas. Os arquivos CSV podem ser facilmente importados e exportados usando programas que armazenam dados em tabelas. O arquivo CSV deve ser colocado em uma pasta intitulada 'csv', localizada no diretório raiz da aplicação.

### 4.3. Configuração do *script*

O caminho do arquivo CSV deve ser especificado no arquivo `newPdf.js`, que é o arquivo de *script* responsável por gerar os certificados. Após essa configuração inicial, as variáveis de ambiente no arquivo `.env` devem ser preenchidas. As variáveis são: `TITLE_ACTIVITY` (título da atividade), `SUBJECT` (tema da atividade), `REALIZATION_DATE` (data de realização da atividade) e `TOTAL_HOURS` (total de horas que a atividade fornece no certificado).

### 4.4. Funcionamento do *script*

O *script* inicia lendo as variáveis de ambiente do arquivo `.env`, as quais contêm informações sobre a atividade e a configuração para a descryptografia da imagem do certificado. Em seguida, é chamada a função `getInfoCSV()` para ler os arquivos CSV e extrair as informações inseridas no formulário. A função `getInfoCSV()` percorre cada arquivo CSV especificado na variável de ambiente `CSV_FILE_PATH_ARRAY`. Para cada arquivo, a função lê o conteúdo usando a biblioteca `csvtojson` e adiciona as informações de cada participante a um array. Caso as informações de um participante já constem no array, suas horas são atualizadas. Caso contrário, a função adiciona um novo objeto ao array com as informações do participante.

Após as informações dos participantes serem coletadas, o *script* chama a função `getImage()` para descryptografar a imagem do certificado. Esta função lê o arquivo de imagem criptografado definido na variável de ambiente `PNG_FILE_PATH`, extrai o vetor de inicialização da imagem criptografada e cria um objeto de decifração com a chave e o algoritmo especificados nas variáveis de ambiente. Na sequência, o *script* usa este objeto para decifrar a imagem.

Com as informações dos participantes e a imagem do certificado prontas, o *script* então percorre as informações de cada participante no array. Caso as horas do participante sejam maiores ou iguais ao valor especificado na variável de ambiente `MIN_HOURS_FOR_CERTIFICATE`, é feita uma chamada à função `MakePDF()` para gerar o certificado. A função `MakePDF()` cria um novo documento PDF usando a biblioteca `jsPDF`, desenhando a imagem do certificado com o auxílio da biblioteca `canvas`. Em seguida, a função adiciona texto ao documento com as informações do participante e da atividade. O documento PDF é salvo em um arquivo na pasta *certificados*. Por fim, o *script* cria uma lista com os e-mails dos participantes da atividade e a salva em um arquivo chamado `email.txt`. Esta lista é utilizada para enviar os certificados aos participantes.

## 5. Resultados obtidos

A implementação do gerador de certificados automatizado trouxe resultados significativos. Através do uso do gerador, foi possível transformar dados de um arquivo CSV em certificados individuais de forma rápida e precisa. O gerador pode ser reutilizado para qualquer outra atividade ou oficina realizada pelo grupo PET-SI.

Com a aplicação, os estudantes passaram a receber seus certificados de maneira mais rápida e isenta de erros de digitação. Além disso, o grupo PET passou a dispor de mais tempo e recursos para se concentrar em outras responsabilidades igualmente relevantes, o que culminou em uma melhoria na qualidade do Programa.

## 6. Conclusão

O desenvolvimento da aplicação de emissão automatizada de certificados foi fundamental tanto para os membros do grupo PET-SI quanto para os alunos da UFSM que se engajaram nas atividades ofertadas. Com a otimização do processo, o grupo conseguiu poupar tempo e esforços, os quais eram necessários na confecção manual e individual dos certificados. Além disso, a ferramenta emissão de certificados anula totalmente a possibilidade de erros de digitação dos nomes, e-mails ou matrículas por parte dos membros do grupo PET na criação dos certificados. Os erros apenas ocorrem nos casos em que o próprio aluno escreve incorretamente alguma informação no formulário. A aplicação de emissão de certificados ainda não está hospedada em nenhum domínio, sendo acessível apenas localmente pelos membros do grupo PET-SI, os quais que têm acesso ao repositório onde o código da aplicação é armazenado.

## Referências

- (2019). Cryptojs documentation. <https://cryptojs.gitbook.io/docs/>. Acessado em: 08 de Agosto de 2023.
- (2022). Curso de javascript. <https://www.cursoemvideo.com/curso/javascript/>. Acessado em: 05 de Julho de 2023.
- Cassanego, S. R. (2020). Gerador de certificados - office interop word. <https://sites.google.com/inf.ufsm.br/pet-si/reda%C3%A7%C3%B5es/2020/gerador-de-certificados-office-interop-word>. Acessado em: 07 de Outubro de 2023.
- Hall, J. (2019). jsPDF documentation. <https://artskydj.github.io/jsPDF/docs/jsPDF.html>. Acessado em: 08 de Agosto de 2023.